Docket No.: 64965-126 (formerly 52352-317) PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of :

Jacques WONG, et al. :

Serial No.: 09/517,518 : Group Art Unit: 2825

Filed: March 02, 2000 : Examiner: Annette M. THOMPSON

For: BOTTOM-UP APPROACH FOR SYNTHESIS OF REGISTER TRANSFER LEVEL (RTL) BASED DESIGN (As Amended)

REPLY BRIEF

Commissioner for Patents
Washington, DC 20231

Sir:

This Reply Brief is submitted in response to the Examiner's Answer mailed December 2, 2002 (earliest mailing date).

THE ARGUMENT

It is again submitted that the rejection of claims 1-3 and 5-17 by the Examiner is predicated upon numerous inaccurate factual determinations. That is, it is submitted that Dupenloup does not disclose or suggest each and every limitation of the claimed invention, as is required under a rejection under 35 U.S.C. §102(e), and that the Examiner has repeatedly misconstrued the language of the Dupenloup reference. An analysis of these inaccurate factual determinations does not require any verbatim or word-for-word equivalence test, but merely a straightforward review of the plain language of the Dupenloup reference in context.

A. **Dupenloup Does Not Anticipate Claims 1 and 12**

Dupenloup does not teach or suggest a method of synthesizing a register transfer level based design of a system including at least the steps of determining a plurality of sub-modules of a top level system, determining individual time budgets for each sub-module based on timing requirements of the top-level system, **synthesizing** gate-level designs of the sub-modules based on the determined time budgets for the individual sub-modules, and **testing** the gate-level designs for conformance with gate-level design requirements of the individual sub-modules, then integrating the gate-level designs of the individual sub-modules to form a top level design, as recited in independent claim 1 and similarly recited in dependent claim 12.

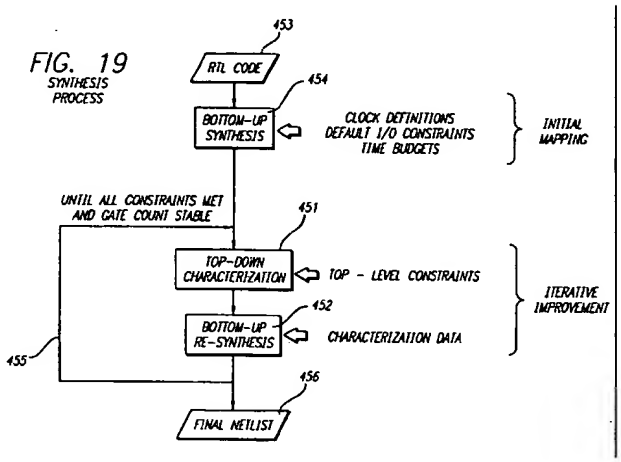
In denying patentability to a claimed invention based upon prior art, **the Examiner must point to "page and line" of a reference wherein each feature of a claimed invention is asserted to reside**. *In re Rijckaert*, 9 F.3d 1531, 28 USPQ2d 1955 (Fed. Cir. 1993). Indeed, as recently held by the Honorable Board of Patent Appeals and Interferences, the Examiner must not only point to "column and line of each relevant prior art reference", but must also explain how one having ordinary skill in the art would have interpreted each of the relied upon portions of the cited references. *Ex parte Gambogi,* 62 USPQ2d 2019 (BPAI 2001). That burden has not been discharged.

Factual Error

Dupenloup does not teach or suggest **testing** the gate-level designs for conformance with gate-level design requirements of the individual sub-modules, then integrating the gate-level designs of the individual sub-modules.

To the contrary, the Examiner has repeatedly misconstrued the plain language of Dupenloup to imply that Dupenloup teaches testing of gate-level designs of sub-modules using passages that are not remotely directed to such activity. For example, the first erroneous reference that the Examiner

provides (see, page 3 of Examiner's Answer) cites col. 41, lines 1-13 and Fig. 19 as disclosing **testing** gate-level designs of individual sub-modules, then integrating the gate-level designs of the individual sub-modules. However, a review of Fig. 19 reveals no manner of testing/verification whatsoever.



Further, an accurate analysis of what Dupenloup actually discloses at cited col. 41, lines 1-13 requires a reading of the cited passage in context. Accordingly, the entire cited passage is reproduced below:

> "1. Bottom-Up **Synthesis**
>
> Referring to FIG. 14, bottom-up **synthesis** starts from leaf modules A 390, B 391, and C 392 in the design hierarchy as illustrated. Those modules are **synthesized**, and a "don't-touch" attribute is set on them. **Synthesis** then proceeds with modules that are located one level up in the design hierarchy, module D 393, and the process continues until the root module E 394 of the design is reached. Because of "don't touch" attributes, lower level modules are considered as non-modifiable cells and are only integrated into upper levels. This dramatically reduces the complexity of **synthesis**." {emphasis added}

As clearly demonstrated by the plain language of Dupenloup, the Examiner's cited language refers to a synthesis step, not a testing step. Thus, the Examiner has used the plain language of Dupenloup, directed solely to logic synthesis, to infer that Dupenloup discloses testing of individual sub-modules. However, **synthesis is not testing**. Accordingly, the cited passages of Dupenloup do

not disclose or suggest testing of individual sub-modules.

Next, the Examiner's Answer cites Dupenloup's Abstract to support the contention that Dupenloup discloses testing of individual sub-modules, Appellants respond by reproducing the Abstract in context below:

> "A method of generating **synthesis** scripts to synthesize integrated circuit (IC) designs in RTL level description into gate-level description comprising the steps of identifying hardware elements in the RTL code, determining key pins for each of said identified hardware elements, extracting design structure and hierarchy from the RTL code, generating script to cause a logic **synthesis** tool to apply bottom-up **synthesis** to modules and sub-modules of the IC design, generating script to cause a logic synthesis tool to apply top-down characterization to modules and sub-modules of the IC design and generating script to cause a logic **synthesis** tool to repeat said bottom-up and said top-down applications until certain predetermined constraints are satisfied." {emphasis added}

Again, as with the passage of column 41, the Abstract makes no reference to any form of testing (or verification) whatsoever, but is directed to synthesis. Accordingly, Dupenloup's Abstract does not disclose or suggest testing of individual sub-modules.

Next, the Examiner's Answer (see, pp. 5-6) cites Fig. 1 and col. 2, lines 19-22, which refers to the "Description of the Related Art" section of Dupenloup, for a purpose that remains unclear to Appellants. However, Appellants assert that Fig. 1 and col. 2, lines 19-22 do not disclose any form of testing (or verification or any other equivalent term) of any sub-module. The cited passage is reproduced below:

> "*At the RTL level, designers must take all key design decisions such as design hierarchy and partitioning, clocking scheme, reset scheme, and locations of registers. All those decisions are contained and reflected in the RTL code.* The RTL code is technology independent, as well as independent from design tools." {emphasis added}

As is apparent by the plain language above, the cited passage relied upon by the Examiner's Answer discloses nothing about testing/verifying sub-modules.

Next, the Examiner's Answer (see, pp. 6-7) cites Fig. 19; col. 43, lines 10-16; col. 43, lines 16-18 and col. 43, lines 21-25 of Dupenloup to support the contention that Dupenloup discloses testing of individual sub-modules. However, as shown above, a review of Fig. 19 reveals that this figure discloses no testing (verification, etc.) whatsoever, but is limited to a synthesis process. Further, for an accurate analysis of what Dupenloup actually discloses requires a reading of the cited passage in context, the entire paragraph (col. 43, lines 6-25) containing the cited passage is reproduced below:

> "This process implements the most powerful **synthesis** and optimization methodology that has been available so far for large designs. The interactive improvement process is illustrated in FIG. 19. *Referring to FIG. 19, the interactive improvement process begins with initial mapping utilizing bottom-up* ***synthesis technique*** *450 with each module being assigned default constraints, time budgets, and clock definitions.* After the initial mapping is completed, top-down characterization 451 is performed. *Top-down characterization provides constraints, time budgets, and other information required to be met by each of the modules.* Then, the constraints determined by the characterization step are used to **resynthesize** each of the modules using bottom-up **resynthesis** technique 452. *The top-down characterization step 451 and bottom-up* ***resynthesis*** *452 steps are iterated 455 until all constraints are met by each of the modules being* ***synthesized*** *and gate count for each of the modules are stable.* Finally, the net list is produced 456." {emphasis added}

As clearly demonstrated by the plain language of Dupenloup, the Examiner's cited language refers to a synthesis step, not a testing step. Thus, the Examiner has again used language directed solely to logic synthesis to infer that Dupenloup discloses testing of individual sub-modules. **Synthesis is not testing.** Accordingly, the cited passages of Dupenloup do not disclose or suggest testing of individual sub-modules.

It is apparent by the plain language of every passage and figure cited in the Examiner's Answer that Dupenloup fails to identically disclose all the requisite claim limitations. To the contrary, the Examiner has merely pointed to certain statements of the specification, as if they showed the claimed step at issue, misconstruing the plain language of the cited reference. While the Examiner puts much

emphasis on the term "verification" to make her rejection, the Examiner nonetheless fails to show where Dupenloup uses the word "verification" (or any derivative or equivalent term) in the context of testing individual sub-modules. In fact, while the terms "test", "testing", "verification" and "verify" are used repeatedly in Dupenloup; there is no instance where Dupenloup discusses any form of testing/verification in the context of individual sub-modules. Accordingly, the Examiner has not discharged her burden to show "page and line" of a reference wherein each feature of a claimed invention is asserted to reside. *In re Rijckaert,* 9 F.3d 1531. **Facts** are required. *In re Lee,* 277 F.3d 1338.

Dupenloup Does Not Anticipate Claims 2-3 and 10-11

Dupenloup does not teach or suggest a method of synthesizing an RTL-based design of a system including generating gate-level netlists for the gate-level designs of each of the sub-modules, and integrating the gate-level designs of the individual sub-modules, as recited in independent claim 10 and dependent claim 2.

Factual Error

While the Examiner's Answer cites col. 1, lines 34-35 and col. 1, lines 45-49, which refers to Dupenloup's "Description of the Related Art" section, Appellants contest this assertion by reproducing the cited passages below:

> "The IC design, as expressed by the RTL code, is then synthesized to generate a gate-level description, or a netlist. This is referred to by the reference number 106 of FIG. 1. Synthesis is the step taken to translate the architectural and functional descriptions of the design, represented by RTL code, to a lower level of representation of the design such as a logic-level and gate-level descriptions. ...
>
> ... This is because, during the synthesis process, the synthesis tool uses a given technology library, 108 of FIG. 1, to map the technology independent RTL code into technology dependent gate-level netlists."

As is apparent by the plain language of the text above, the **cited passages make no reference to generating netlists for individual sub-modules**. Further, Appellants contest that Dupenloup's "Description of the Related Art" section even refers to synthesizing a design using sub-modules as, contrary to the Examiner's remarks regarding col. 2, lines 19-22, a hierarchical design or a design capable of partitioning does not require hierarchical synthesis, but merely a compiler capable of translating hierarchical/partitioned design expressed in a high-level language into RTL code.

The Examiner's next asserted passage at col. 41, lines 21-25 again neglects to make reference to generating netlists for individual sub-modules, as is shown by the plain language of the passage that the Examiner cites, reproduced below:

> "Top-down characterization consists in calculating all I/O conditions and constraints of each of the modules and the sub-modules in a hierarchical design."

While the cited passage does refer to top-down characterization, it does not disclose generating netlists for individual sub-modules. Appellants also assert that Fig. 25 and col. 41, lines 6-13, also cited by the Examiner, does not disclose generating netlists for individual sub-modules. While the Examiner appears to imply that making a particular low-level module non-modifiable suggests generating netlists for individual sub-modules, Appellants respectfully reply by asserting that freezing a particular module does not require netlist generation, and the Examiner provided no documented evidence to suggest that it does.

Dupenloup Does Not Anticipate Claims 5 and 13

Appellants assert that claims 5 and 13 are patentable by virtue of their dependency as well as for the additional features they recite, including performing static timing analysis on individual sub-modules for conformance with timing requirements for individual sub-blocks, as recited in dependent

claims 5 and 13.

Factual Error

While the Examiner asserts that Dupenloup discloses performing static timing analysis on individual sub-modules at col. 12, lines 43-48 42 and col. 13, lines 33-36. While the passage does refer to static timing analysis, i.e., checking clock domain interfaces, the cited passage does not disclose performing static timing analysis on individual sub-modules. In fact, a review of the cited passage, surrounding text and related Fig. 4 (Fig. 3A cited by the Examiner relates to col. 11, line 55 to col. 12, line 8) reveals that at most this passage refers to checking clock domains for various logic elements with no regard to sub-modules.

Dupenloup Does Not Anticipate Claims 6-8 and 14-16

Appellants assert that claims 6-8 and 14-16 are patentable by virtue of their dependency as well as for the additional features they recite, including performing static timing analysis on individual sub-modules for conformance with timing requirements for individual sub-blocks, as recited in dependent claims 5 and 13.

Factual Error

While the Examiner now asserts Fig 19 supporting the assertion that Dupenloup discloses static timing analysis on individual sub-modules, Appellants cite Fig. 19 to demonstrate that Fig. 19 does not even refer to the timing requirements for individual sub-modules. Accordingly, the Examiner has not met the requisite burden of proof to support a claim of anticipation under 35 U.S.C. §102(e).

E. **Dupenloup Does Not Anticipate Claims 9 and 17**

Appellants assert that claims 9 and 17 are patentable by virtue of their dependency as well as for the additional features they recite, including a step of verifying conformance of the gate-level designs that includes performing dynamic simulations on the gate-level designs (of the individual sub-modules), as recited in dependent claims 6 and 14.

While the Examiner now asserts that "the process of characterization disclosed in Dupenloup at column 41, lines 16-52 involves dynamic simulation", Appellants point to the plain language of the cited passage, which does not even refer to simulations.

Further, while the Examiner raises new issues using Appellants specification citing page 4, lines 24-26, Appellants assert that using any secondary reference, including Appellants' own specification, is impermissible in any claim of anticipation under 35 U.S.C. §102(e).

F. **Conclusion:**

Based upon the arguments submitted ***supra.***, it is respectfully submitted that the record does not establish that any of the appealed claims are anticipated under 35 U.S.C. §102. Appellants, therefore, respectfully solicit reversal of the imposed rejections under 35 U.S.C. §102 as erroneous.

IX. PRAYER FOR RELIEF

To the extent necessary, a petition for an extension of time under 37 C.F.R. 1.136 is hereby made. Please charge any shortage in fees due in connection with the filing of this paper, including extension of time fees, to Deposit Account 500417 and please credit any excess fees to such deposit account.

Respectfully submitted,

MCDERMOTT, WILL & EMERY

B. Y. Mathis
Registration No. 44,907

600 13th Street, N.W.
Washington, DC 20005-3096
(202) 756-8000 BYM/JAH:kap
Date:
Facsimile: (202) 756-8087